

Отчёт по лабораторной работе №5

**Дискреционное разграничение прав в Linux. Исследование влияния
дополнительных атрибутов**

Кузьмин Артем

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
2.1	Подготовка	4
2.2	Изучение механики SetUID	5
2.3	Исследование Sticky-бита	10
3	Выводы	12
	Список литературы	14

List of Figures

2.1	подготовка к работе	5
2.2	программа simpleid	6
2.3	результат программы simpleid	7
2.4	программа simpleid2	7
2.5	результат программы simpleid2	8
2.6	программа readfile	9
2.7	результат программы readfile	10
2.8	исследование Sticky-бита	12

1 Цель работы

Изучение механизмов изменения идентификаторов, применения SetUID и Stickyбитов.
Получение практических навыков работы в консоли с дополнительными атрибутами.
Рассмотрение работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

2 Выполнение лабораторной работы

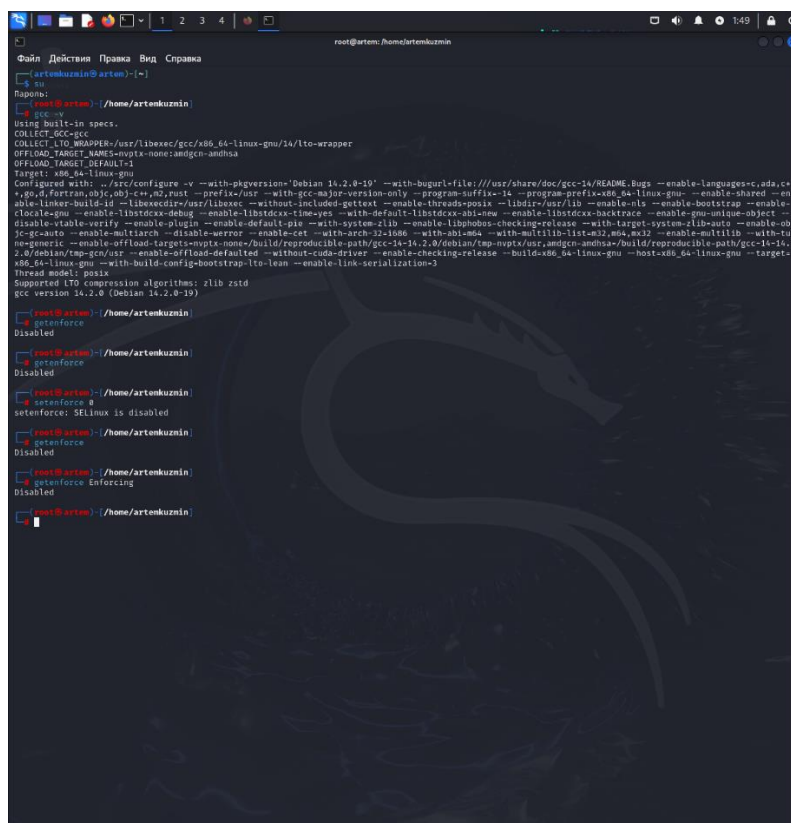
2.1 Подготовка

1. Для выполнения части заданий требуются средства разработки приложений.

Проверили наличие установленного компилятора gcc командой `gcc -v`:
компилятор обнаружен.

2. Чтобы система защиты SELinux не мешала выполнению заданий работы, отключили систему запретов до очередной перезагрузки системы командой `setenforce 0`:

3. Команда `getenforce` вывела Permissive:



```
root@artem: /home/artemkuzmin
[artemkuzmin@artem]~
$ su
[artem@artem]~/home/artemkuzmin
$ gcc
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-linux-gnu/14/lto-wrapper
OFFLOAD_TARGET_NAMES=mvptx-none;amdgcn-amdhsa
OFFLOAD_TARGET_DEFAULT=1
Target: x86_64-linux-gnu
Configured with: ./src/configure -v --with-pkgversion='Debian 14.2.0-19' --with-bugurl=file:///usr/share/doc/gcc-14/README.Bugs --enable-languages=c,ada,c++,go,d,fortran,objc,objc++,m2,must --prefix=/usr --with-gcc-major-version-only --program-suffix=-14 --program-prefix=x86_64-linux-gnu- --enable-shared --enable-linker-build-id --libexecdir=/usr/libexec --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-rtls --enable-bootstrap --enable-cloCALE-gnu --enable-libstdcxx-debug --enable-libstdcxx-time-yes --with-default-libstdcxx-abi=new --enable-libstdcxx-backtrace --enable-gnu-unique-object --disable-able-verify --enable-plugin --enable-default-pie --with-system-zlib --enable-libphobos-checking-release --with-target-system-zlib=auto --enable-objc-c++-auto --enable-multiarch --disable-werror --enable-cet --with-arch=32-i386 --with-abi=x86_64 --with-multilib-list=m32,m64,mx16 --enable-multilib --with-tune=generic --enable-offload-targets=mvptx-none,build/reproducible-path/gcc-14-14.2.0/debian/tmp-gcc/usr --enable-offload-defaulted --without-cuda-driver --enable-checking-release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu --with-build-config=bootstrap-lto-lean --enable-link-serialization=3
Thread model: posix
Supported LTO compression algorithms: zlib zstd
gcc version 14.2.0 (Debian 14.2.0-19)

[artem@artem]~/home/artemkuzmin
$ getenforce
Disabled

[artem@artem]~/home/artemkuzmin
$ getenforce
Disabled

[artem@artem]~/home/artemkuzmin
$ setenforce 0
setenforce: SELinux is disabled

[artem@artem]~/home/artemkuzmin
$ getenforce
Disabled

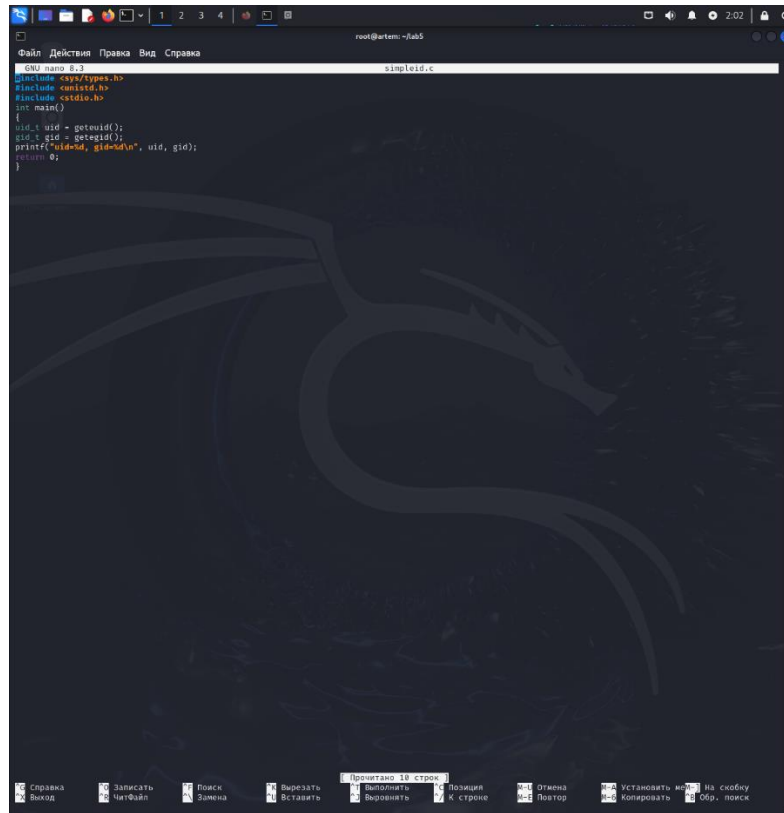
[artem@artem]~/home/artemkuzmin
$ getenforce Enforcing
Disabled

[artem@artem]~/home/artemkuzmin
$
```

Figure 2.1: подготовка к работе

2.2 Изучение механики SetUID

1. Вошли в систему от имени пользователя guest.
2. Написали программу simpleid.c.

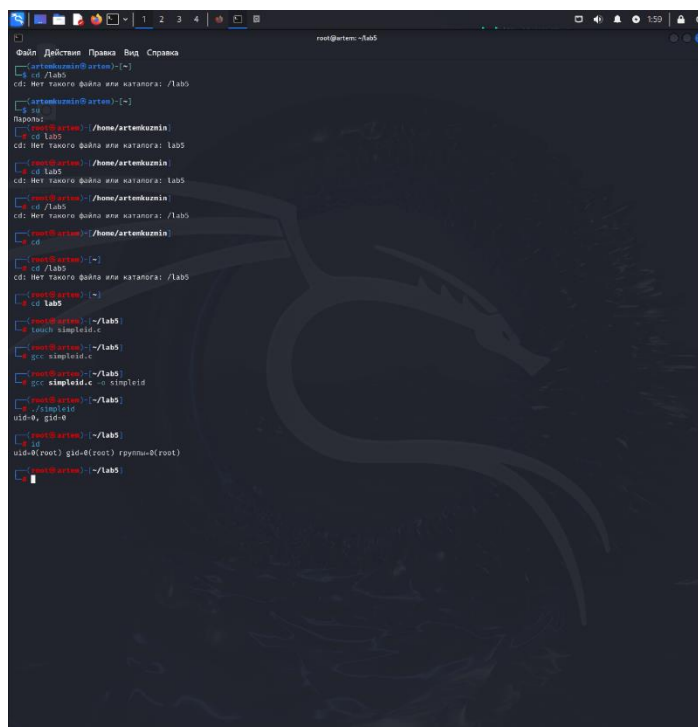


```
root@artem: ~# cat simpleid.c
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>

int main()
{
    uid_t uid = getuid();
    gid_t gid = getsid();
    printf("uid=%d, gid=%d\n", uid, gid);
    return 0;
}
```

Figure 2.2: программа simpleid

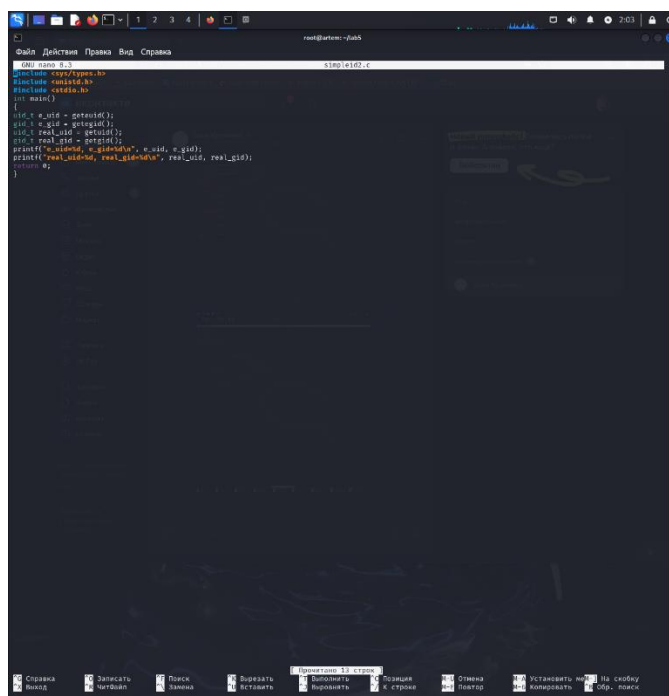
3. Скомпилировали программу и убедились, что файл программы создан: `gcc simpleid.c -o simpleid`
4. Выполнили программу simpleid командой `./simpleid`
5. Выполнили системную программу `id` с помощью команды `id`. `uid` и `gid` совпадает в обеих программах



```
root@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
ls
ls: cannot access 'ls': No such file or directory
artem@artem:~/lab5
cd /home/artemkuzmin
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
touch simpleid.c
artem@artem:~/lab5
gcc simpleid.c
artem@artem:~/lab5
gcc simpleid.c -o simpleid
artem@artem:~/lab5
./simpleid
uid=0, gid=0
artem@artem:~/lab5
id
id=0(root) gid=0(root) rgroup=0(root)
artem@artem:~/lab5
```

Figure 2.3: результат программы simpleid

6. Усложнили программу,добавив вывод действительных идентификаторов.



```
root@artem:~/lab5
ls
ls: cannot access 'ls': No such file or directory
artem@artem:~/lab5
cd /home/artemkuzmin
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
cd /lab5
cd: Her Yakovlev Gubina aka katarina: /lab5
artem@artem:~/lab5
touch simpleid2.c
artem@artem:~/lab5
gcc simpleid2.c
artem@artem:~/lab5
gcc simpleid2.c -o simpleid2
artem@artem:~/lab5
./simpleid2
uid=0, gid=0
artem@artem:~/lab5
id
id=0(root) gid=0(root) rgroup=0(root)
artem@artem:~/lab5
```

Figure 2.4: программа simpleid2

7. Скомпилировали и запустили simpleid2.c:

```
gcc simpleid2.c -o simpleid2
```

```
./simpleid2
```

8. От имени суперпользователя выполнили команды:

```
chown root:guest /home/guest/simpleid2 chmod u+s
```

```
/home/guest/simpleid2
```

9. Использовали su для повышения прав до суперпользователя

10. Выполнили проверку правильности установки новых атрибутов и смены владельца файла simpleid2:

```
ls -l simpleid2
```

11. Запустили simpleid2 и id:

```
./simpleid2 id
```

Результат выполнения программ теперь немного отличается

12. Проделали тоже самое относительно SetGID-бита.

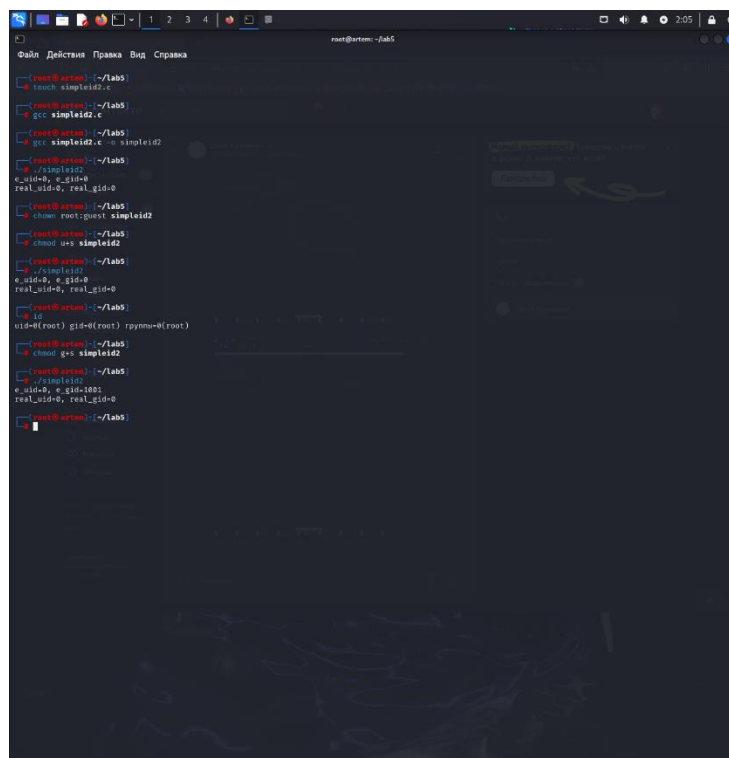


Figure 2.5: результат программы simpleid2

13. Написали программу readfile.c

```
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
#include <fcntl.h>

int main(int argc, char* argv[])
{
    unsigned char buffer[16];
    size_t bytes_read;
    int i;

    int fd=open(argv[1], O_RDONLY);
    do
    {
        bytes_read=read(fd, buffer, sizeof(buffer));
        for (i=0; i<bytes_read; ++i)
            printf("%c", buffer[i]);
    }
    while (bytes_read == (buffer));
    close (fd);
    return 0;
}
```


Figure 2.6: программа readfile

14. Откомпилировали её.

```
gcc readfile.c -o readfile
```

15. Сменили владельца у файла readfile.c и изменили права так, чтобы только суперпользователь (root) мог прочитать его, а guest не мог.

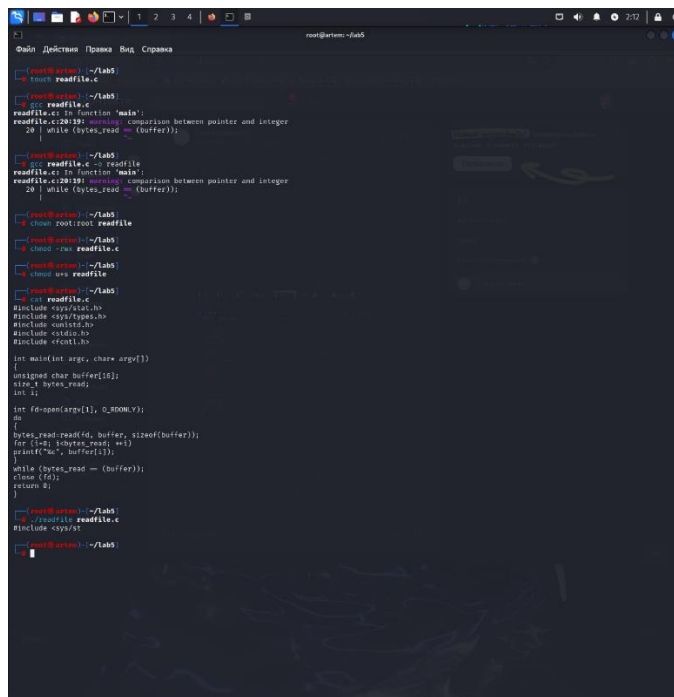
```
chown root:guest /home/guest/readfile.c chmod 700  
/home/guest/readfile.c
```

16. Проверили, что пользователь guest не может прочитать файл readfile.c.

17. Сменили у программы readfile владельца и установили SetU'D-бит.

18. Проверили, может ли программа readfile прочитать файл readfile.c

19. Проверили, может ли программа readfile прочитать файл /etc/shadow



```
root@ubuntu:~/lab5# touch readfile.c
root@ubuntu:~/lab5# gcc readfile.c
root@ubuntu:~/lab5# chown root:root readfile
root@ubuntu:~/lab5# chmod 700 readfile
root@ubuntu:~/lab5# ./readfile
root@ubuntu:~/lab5#
```

Figure 2.7: результат программы readfile

2.3 Исследование Sticky-бита

1. Выяснили, установлен ли атрибут Sticky на директории /tmp:

```
ls -l / | grep tmp
```

2. От имени пользователя guest создали файл file01.txt в директории /tmp со словом test:

```
echo "test" > /tmp/file01.txt
```

3. Просмотрели атрибуты у только что созданного файла и разрешили чтение и запись для категории пользователей «все остальные»:

```
ls -l /tmp/file01.txt chmod o+rw
```

```
/tmp/file01.txt ls -l /tmp/file01.txt
```

Первоначально все группы имели право на чтение, а запись могли осуществлять все, кроме «остальных пользователей».

4. От пользователя (не являющегося владельцем) попробовали прочитать файл /file01.txt:

```
cat /file01.txt
```

5. От пользователя попробовали дозаписать в файл /file01.txt слово test3 командой:

```
echo "test2" >> /file01.txt
```

6. Проверили содержимое файла командой: cat /file01.txt

В файле теперь записано:

```
Test
```

```
Test2
```

7. От пользователя попробовали записать в файл /tmp/file01.txt слово test4, стерев при этом всю имеющуюся в файле информацию командой. Для этого воспользовалась командой `echo "test3" > /tmp/file01.txt`

8. Проверили содержимое файла командой

```
cat /tmp/file01.txt
```

9. От пользователя попробовали удалить файл /tmp/file01.txt командой `rm /tmp/file01.txt`, однако получила отказ.
10. От суперпользователя командой выполнили команду, снимающую атрибут t (Sticky-бит) с директории /tmp:

```
chmod -t /tmp
```

Покинули режим суперпользователя командой `exit`.

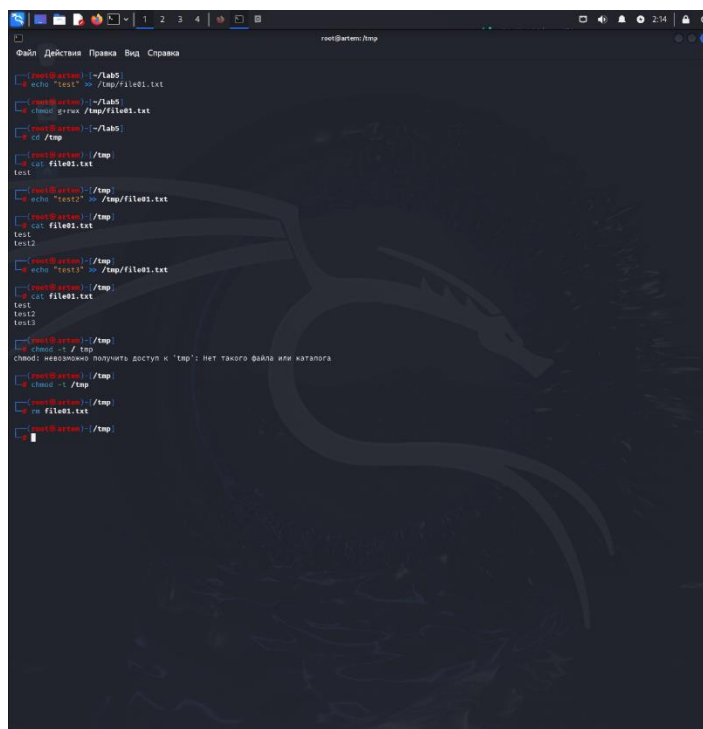
11. От пользователя проверили, что атрибута t у директории /tmp нет:

```
ls -l / | grep tmp
```

12. Повторили предыдущие шаги. Получилось удалить файл
13. Удалось удалить файл от имени пользователя, не являющегося его владельцем.
14. Повысили свои права до суперпользователя и вернули атрибут t на директорию /tmp :

```
su chmod +t /tmp
```

```
exit
```



```
root@artem:/tmp
root@artem: ~/# cd /tmp
root@artem: /tmp/# echo "test" > /tmp/file01.txt
root@artem: /tmp/# ls -l /tmp
total 4
-rw-rw-rw- 1 root root 11 Nov 14 12:14 file01.txt
root@artem: /tmp/# cd /tmp
root@artem: /tmp/# cat file01.txt
test
root@artem: /tmp/# echo "test2" > /tmp/file01.txt
root@artem: /tmp/# cat file01.txt
test2
root@artem: /tmp/# echo "test3" > /tmp/file01.txt
root@artem: /tmp/# cat file01.txt
test3
root@artem: /tmp/# rm file01.txt
rm: cannot remove 'file01.txt': Operation not permitted
root@artem: /tmp/#
```

Figure 2.8: исследование Sticky-бита

3 Выводы

Изучили механизмы изменения идентификаторов, применения SetUID- и Stickyбитов. Получили практические навыки работы в консоли с дополнительными атрибутами. Также мы рассмотрели работу механизма смены идентификатора процессов пользователей и влияние бита Sticky на запись и удаление файлов.

Список литературы

1. КОМАНДА CHATTR В LINUX
2. chattr